Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently Amended) A method of managing different types of events in a distributed computing system, having an event engine, including the steps of:

i. providing one or more intelligent agents for receiving a first event and converting the first event into a standard format, and inputting the first event into the engine;

ii. the engine extracting a rule to be applied to the first event from a rules database wherein identification information within the rule identifies the first event;

iii. the engine holding the first event for an expiration of a specified interval;

iv. before the expiration of the specified interval, receiving at least one subsequent event from an intelligent agent, converting the subsequent event into a standard format and inputting the subsequent event into the engine;

v. the engine identifying the subsequent event using identification information within the rule;

vi. waiting for identical subsequent events;

vii. if, during the specified interval, a predetermined number of identical subsequent events are received, the engine creating and outputting a new event with an attribute that indicates the number of identical subsequent events that were received;

viii. inputting the new event into the engine;

ix. the engine extracting a second rule to be applied to the new event from a rules database wherein identification information within the second rule identifies the new event;

x. determining whether the number of identical subsequent events exceeds a threshold;

xi creating a threshold event indicating whether the number of identical subsequent events exceeds the threshold; and

xii outputting the threshold event to a user, wherein the threshold event is output to the user in a format independent of a source of the event comprising:

a severity code;

predefined actions available to the user; and

event-specific instructions for the user, wherein information resulting from execution of the predefined actions or event-specific instructions is stored in a central database to automate the execution of the predefined actions or event-specific instructions upon recurrence of the threshold event.

2. (Previously Presented) A method as claimed in claim 1 wherein the first event and the subsequent event originate from any of a set of a network, an application or an operating system residing on the distributed computing system, and hardware.

3. - 4. (Cancelled)

5. (Previously Presented) A method as claimed in claim 1, wherein the identification information includes:

i. an attribute;

ii. an operator; and

iii. a value.

6. (Original) A method as claimed in claim 5 wherein the specified interval is time.

7. (Currently Amended) A method of managing different types of events in a distributed computing system, having an event engine, including the steps of:

i. providing one or more intelligent agents configured for receiving at least one event specifying a type of server failure and converting the event into a standard format, and inputting the event into the engine;

ii. the engine extracting a rule to be applied to the event from a rules database wherein identification information within the rule identifies the event;

iii. the engine creating and outputting a new event indicating a server failure has occurred;

iv. inputting the new event into the engine;

v. the engine extracting a second rule to be applied to the new event from the rules database wherein identification information within the second rule identifies the new event;

vi. the engine holding the new event for the expiration of a specified interval;

vii. before the expiration of the specified interval, receiving at least one subsequent event from an intelligent agent, converting the subsequent event into a standard format and inputting the subsequent event into the engine;

viii. the engine identifying the subsequent event using identification information within the second rule; and

ix. wherein, if the subsequent event indicates a server restoration has occurred, outputting the subsequent event to a user, or if the subsequent event indicates a type of server failure, the subsequent event is discarded, and after the expiration of the specified interval the new event indicating a server failure has occurred is output to the user, wherein the new event indicating a server failure has occurred is output to the user in a format independent of a source of the event comprising:

a severity code;

predefined actions available to the user; and

event-specific instructions for the user, wherein information resulting from execution of the predefined actions or event-specific instructions is stored in a central database to automate the execution of the predefined actions or event-specific instructions upon recurrence of an event indicating a server failure has occurred.

8. (Previously Presented) A method as claimed in claim 7 wherein the event and the subsequent event originate from any of a set of a network, an application, an operating system, and hardware.

9. – 10. (Cancelled)

11. (Previously Presented) A method as claimed in claim 8, wherein the identification information includes:

i. an attribute;

ii. an operator; and

iii. a value.

12. (Original) A method as claimed in claim 11 wherein the specified interval is time.

13. (Previously Presented) A method as claimed in claim 12 wherein the subsequent event is received by a user console.

14. (Currently Amended) A method of managing different types of events in a distributed computing system using a management server, having an event engine, including the steps of:

i. providing one or more intelligent agents for receiving at least one event related to the performance of a network and converting the event into a standard format, and inputting the event into the engine;

ii. the engine extracting a first rule to be applied to the event from a rules database wherein identification information within the first rule identifies the event;

iii. the engine creating and outputting a new event having an attribute set to the type of event related to the performance of the network received;

iv. inputting the new event into the engine;

v. receiving at least one subsequent event indicating a portal service failure event from an intelligent agent, converting the subsequent event into a standard format and inputting the subsequent event into the engine;

vi. the engine extracting a second rule to be applied to the subsequent event from a rules database wherein identification information within the second rule identifies the subsequent event;

vii. the engine holding the subsequent event for an expiration of a specified interval;

viii. before the expiration of the specified interval, receiving the new event; and

ix. the engine creating and outputting an event identifying the cause of the portal service failure, having an attribute set to the attribute of the new event that is set to the type of event related to the performance of the network, wherein the new event identifying the cause of the portal service failure is output to the user in a format independent of a source of the event comprising:

a severity code;

predefined actions available to the user; and

event-specific instructions for the user, wherein information resulting from execution of the predefined actions or event-specific instructions is stored in a central database to automate the execution of the predefined actions or event-specific instructions upon recurrence of the event identifying the cause of the portal service failure.

15. (Previously Presented) A method as claimed in claim 14 wherein the event and the new event originate from any of the set of a network, an application, an operating system, and hardware.

16. – 17. (Cancelled)

18. (Previously Presented) A method as claimed in claim 15, wherein the identification information includes:

i. an attribute;

ii. an operator; and

iii. a value.

19. (Original) A method as claimed in claim 18 wherein the specified interval is time.

20. (Previously Presented) A method a claimed in claim 19 wherein the new event is received by a user console.

21. (Currently Amended) A method of managing different types of events in a distributed computing system including the steps of:

i. receiving a first event indicating a database failure and converting the first event into a standard format;

ii. extracting a rule to be applied to the first event from a rules database wherein identification information within the rule identifies the first event;

iii. the engine holding the first event for an expiration of a specified interval;

iv. if a subsequent event indicating the database is restored is received before expiration of the specified interval, discarding the first event and the subsequent event; and

v. if the subsequent event indicating the database is restored is not received before expiration of the specified interval, outputting the first event indicating database failure to a use, wherein the new event indicating database failure is output to the user in a format independent of a source of the event comprising:

a severity code;

predefined actions available to the user; and

event-specific instructions for the user, wherein information resulting from execution of the predefined actions or event-specific instructions is stored in a central database to automate the execution of the predefined actions or event-specific instructions upon recurrence of an event indicating database failure.

22. (Cancelled)

23. (Currently Amended) A method of managing different types of events in a distributed computing system including the steps of:

i. processing a first event by:

ii. receiving the first event and converting the first event into standard format;

iii. extracting one or more rules which match the event from a rules database;

iv. holding the first event for a specified period of time;

v. receiving at least one subsequent event within the specified period of time;

vi. if the subsequent event is the same type of event as the first event, discarding the subsequent event;

vii. at the end of the specified period of time, creating a new event that indicates the number of subsequent events that were discarded during the specified period of time; and

viii. outputting the first event and the new event to a user, wherein the first event and the new event are output to the user in a format independent of a source of the first event and the new event comprising:

a severity code;

predefined actions available to the user; and

event-specific instructions for the user, wherein information resulting from execution of the predefined actions or event-specific instructions is stored in a central database to automate the execution of the predefined actions or event-specific instructions upon recurrence of the first event and the new event.

24. (Currently Amended) A computer implemented system for managing different types of events in a distributed computing system, embodied in a computer readable medium, including:

i. computer code for providing a plurality of event agents adapted to receive data from a source, to create an event from the data, convert the event into a standard format and to transmit the event to a central event system; and

ii. computer code for providing a central event system including:

a) a rules database adapted to store a plurality of rules, each rule including:

I. identification information specifying to which events the rule relates; and

II. an action for filtering, correlating or consolidating one or more received events wherein the action is one of outputting the event, discarding the event, holding the event, or creating a new event;

wherein, where the action is holding the event the rule further includes:

I. a condition; and

II. a further action wherein the further action is one of outputting the event, discarding the event, holding the event, creating a new event, or creating a new event and transmitting the new event back into the processing engine; and

b) a processing engine adapted to receive events, to extract rules from the rules database, to identify which rules apply to the events using the identification information within the rule, to perform the action specified within the applicable rules, [[and]] to perform the further action specified within the applicable rules when the corresponding condition is satisfied[[.]] and to output the events to a user, wherein the event is output to the user in a format independent of a source of the event comprising:

a severity code;

predefined actions available to the user; and

event-specific instructions for the user, wherein information resulting from execution of the predefined actions or event-specific instructions is stored in a central database to automate the execution of the predefined actions or event-specific instructions upon recurrence of the event.

25. (Previously Presented) A computer implemented system as claimed in claim 24 including one or more user consoles adapted to receive one or more of the events outputted by the central event system.

26. (Previously Presented) A computer implemented system as claimed in claim 25 wherein the source is any one of a set of a database, an application, an operating system, and hardware.

27. (Previously Presented) A computer implemented system as claimed in claim 26 wherein the identification information includes:

i. an attribute;

ii. an operator; and

iii. a value.

28. – 29. (Cancelled)

30. (Previously Presented) Storage media containing software for executing the method of claim 21.

31. – 32. (Cancelled)